

ソフトウェア進化の メタモデル階層の提案

下滝 亜里 青山 幹雄

南山大学 大学院 数理情報研究科

発表のシナリオ

- 研究の背景と課題
- 探索行為としての意思決定
- 意思決定を支援する指針
- ソフトウェア進化のモデル化の要件
- アプローチ
- ソフトウェア進化のメタモデル階層
- 今後の課題
- まとめ

研究の背景と課題

意思決定：選択肢から選択すること

背景

- 開発者は、環境やユーザ要求の変化に対し、適切な**意思決定**を行ってソフトウェアを進化できなければならない

研究目標

- 意思決定を支援し、進化を方向付ける指針の開発

課題

- 意思決定の観点からのソフトウェア進化のモデル化

提案

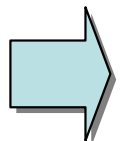
- 3層から成るソフトウェア進化のメタモデル階層
 - 提案モデルを用いて進化の指針を開発

探索行為としての意思決定

意思決定の難しさの要因を、適切な**意思決定結果**を得るための探索の難しさの観点から捉える

● 探索行為としての意思決定

- システムに対する機能・非機能要求を満たすように、意思決定が行われる
- システムの様々な側面，抽象度，粒度の観点からの，相互に依存する多数の意思決定が必要
- 多数の意思決定が必要なため，初めから適切な意思決定結果を得るのは困難
- 適切な意思決定結果を得るために，ある意思決定結果から他の意思決定結果への探索が行われ，**意思決定結果は変化する**
- 意思決定結果の探索は開発者の知識や経験に依存して行われる
- 環境やユーザ要求の変化に対しては，再探索が必要
- 開発者の知識や経験が変化したとき，再探索が行われる



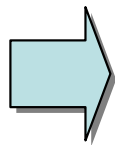
探索や再探索を容易にできるための支援が必要

意思決定を支援する指針

探索済みの意思決定を表現する指針により、新規探索や再探索を支援

- 進化の指針

- ある意思決定結果と他の意思決定結果, その間の変化過程を表現



進化の指針の開発のためには、意思決定結果とその変化過程の観点からのモデル化が必要

意思決定の観点からの進化のモデル化の要件

進化モデルに対する要件を「何を表現できなければならないか」と「どんな表現でなければならないのか」という観点から定義

- **要件(1): 意思決定結果とその変化過程を表現できること**
 - 進化の指針を開発するために、**意思決定結果とその変化過程**を表現できること
- **要件(2): 指針を表現できること**
 - 進化モデルから、指針を抽出できなければならない
- **要件(3): 一般的であること**
 - どんな指針が存在するのか分からないため、進化モデルは、システムの様々な側面、抽象度、粒度での**意思決定結果とその変化過程**を一般的に表現できる必要がある
- **要件(4): シンプルであること**
 - モデルを用いることによる指針の開発を容易にするため
- **要件(5): 形式的であること**
 - 形式的な指針を開発し、ツールによる支援を行うため

モデル化の要件に対するアプローチ

要件(1): 意思決定結果とその変化過程を表現できること

- **意思決定結果と意思決定結果の変化**の観点からモデルが表現すべき対象の要件を詳細化して定義し、モデル化の要素を特定
 - **意思決定結果**に関わる8つの要件
 - **意思決定結果の変化**に関わる9つの要件

要件(2): 指針を表現できること

- 指針を表現するモデル層を導入

要件(3): 一般的であること

- 高い抽象度でのモデル化

要件(4): シンプルであること

- モデル化の要素数を少なくする

要件(5): 形式的であること

- モデル駆動工学におけるメタモデリングのアプローチにより、形式的なモデルを定義
- メタモデリング言語としてMeta-Object Facilityを使用

モデル化の要件：意思決定結果

意思決定結果に関する特性を要件として定義

- **要件(1)**: ソフトウェアは機能や構造, 性質に関わる多数の意思決定結果からなる.
- **要件(2)**: 意思決定結果には, 概念レベルから実体レベルまで様々な抽象度がある.
- **要件(3)**: 意思決定結果には, 細粒度なものから粗粒度まで様々な粒度がある.
- **要件(4)**: 意思決定結果には, 同じ種類のものが存在する.
- **要件(5)**: 意思決定結果間には様々な種類の依存関係がある
- **要件(6)**: 依存関係には, 同じ抽象度間での依存関係と, 異なる抽象度間での依存関係がある
- **要件(7)**: 概念レベルの意思決定結果があるため, 開発者が全ての依存関係をあらかじめ認識することは困難である
- **要件(8)**: 同じ種類の意思決定結果が存在するため, 意思決定結果とその間の依存関係からなる構造には, 同一の構造が存在しうる

モデル化の要件：意思決定結果の変化

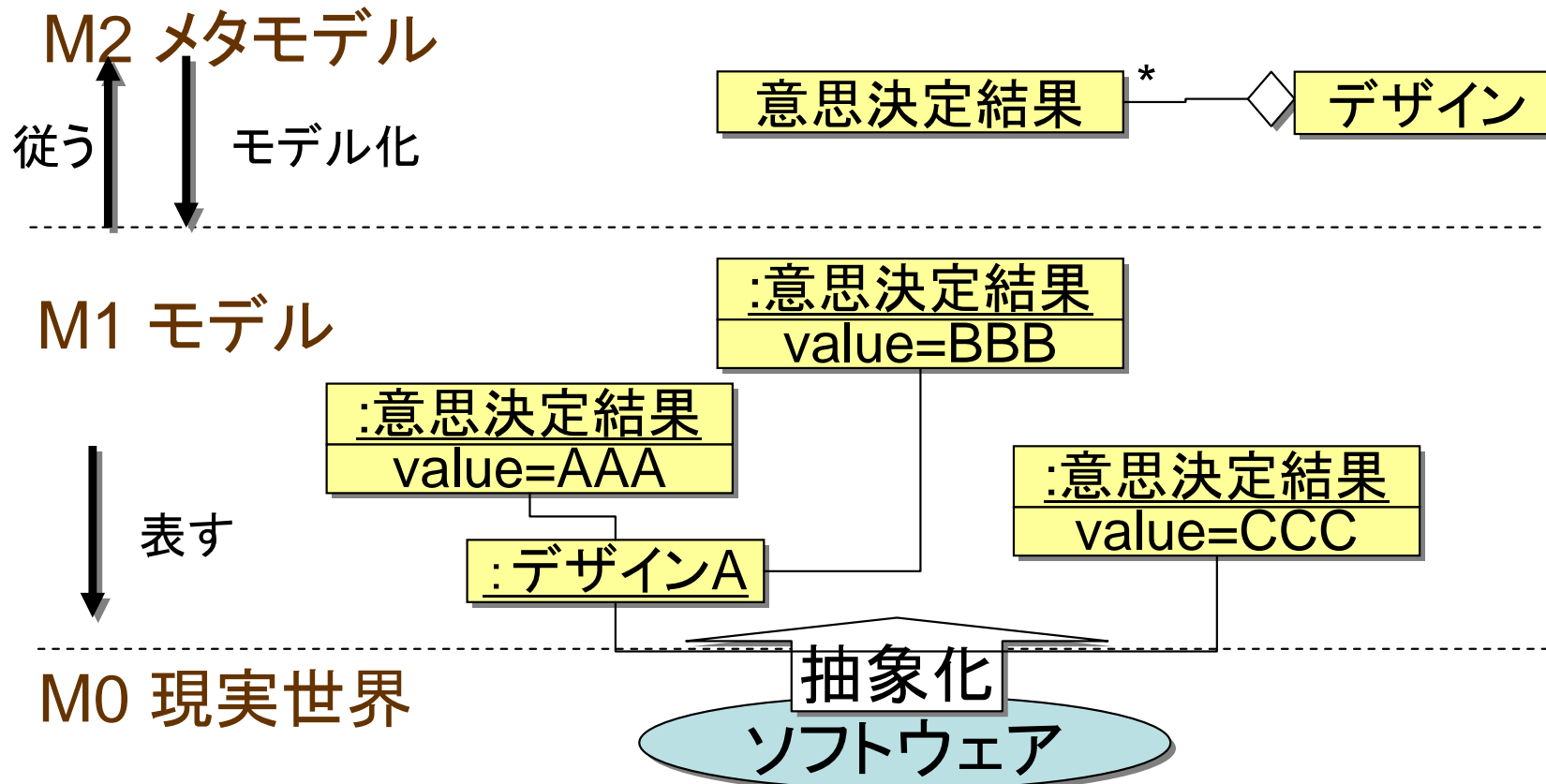
意思決定結果の変化に関する特性を要件として定義

- **要件(a)**: ソフトウェア進化のプロセスは、外発的・内発的な要因の変化がドライバとなり、発生する。
- **要件(b)**: 環境やユーザ要求など外発的な要因の変化に対応できるように、意思決定結果は変化する。
- **要件(c)**: 問題に対する開発者の認識など内発的な要因の変化に伴い、意思決定結果は変化する。
- **要件(d)**: 内発的・外発的要因の変化に伴って、変化する意思決定結果と変化しない意思決定結果が存在する。
- **要件(e)**: 意思決定結果の変化の種類には、新規の意思決定の追加、既存の意思決定の削除、既存の意思決定の変更がある。これらを変更操作と呼ぶ。
- **要件(f)**: 変更操作は、操作対象となる意思決定結果により決まる。
- **要件(g)**: 意思決定結果の変化は、概念レベルから実体レベルの意思決定結果に対する複数の変更操作の組み合わせから成る。
- **要件(h)**: 意思決定結果は、粗粒度から細粒度に向けて詳細化される。
- **要件(i)**: 意思決定結果間の依存関係の変化の種類には、関係の追加、関係の削除、関係の変更がある。

意思決定結果のモデル化(1/4)

要件(1)ソフトウェアは機能や構造, 性質に関わる多数の意思決定結果からなる

- ・どんな側面に関わるかは区別せずに単なる「意思決定結果」要素として抽象化して表現
- ・意思決定結果の集合を表現する要素として「デザイン」を導入

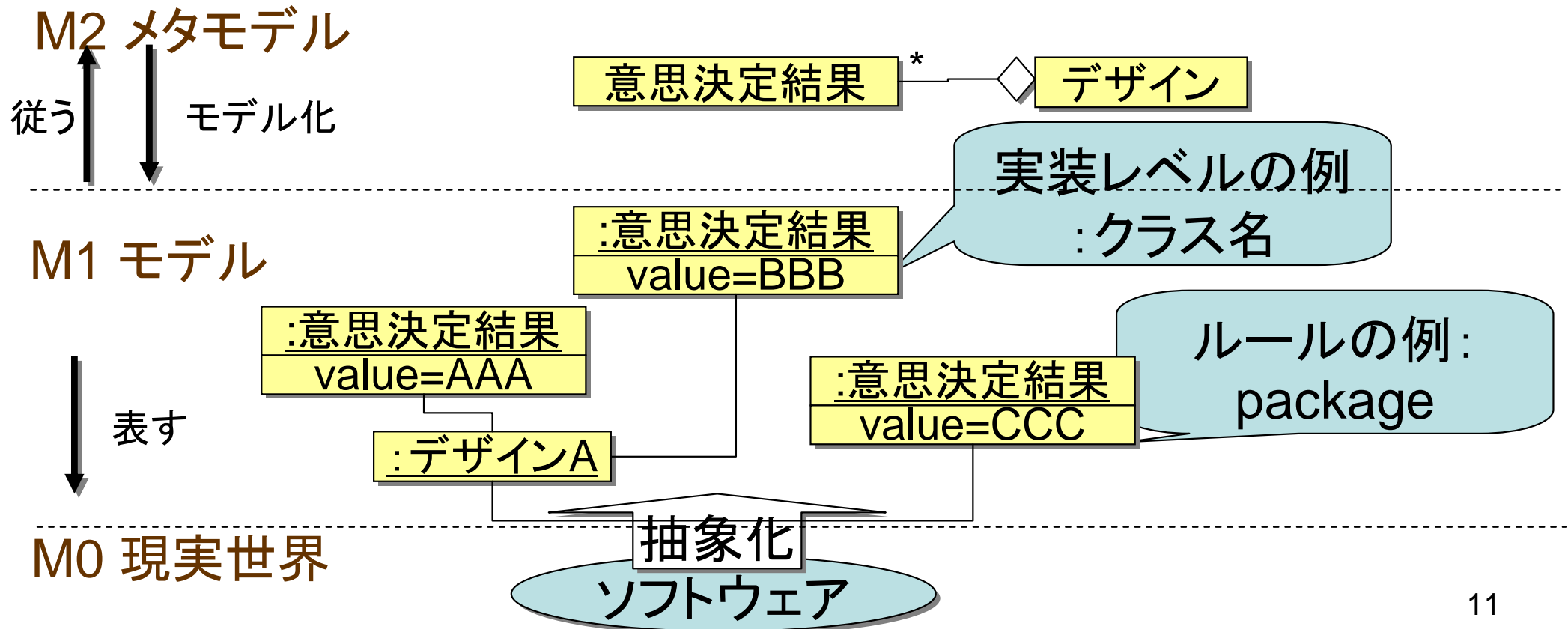


意思決定結果のモデル化(2/4)

要件(2)意思決定結果には, 概念レベルから実体レベルまで様々な抽象度がある.

➤例: 実装レベル, アーキテクチャレベル, 制約やルール.

➡ 抽象度を区別せずにモデル化



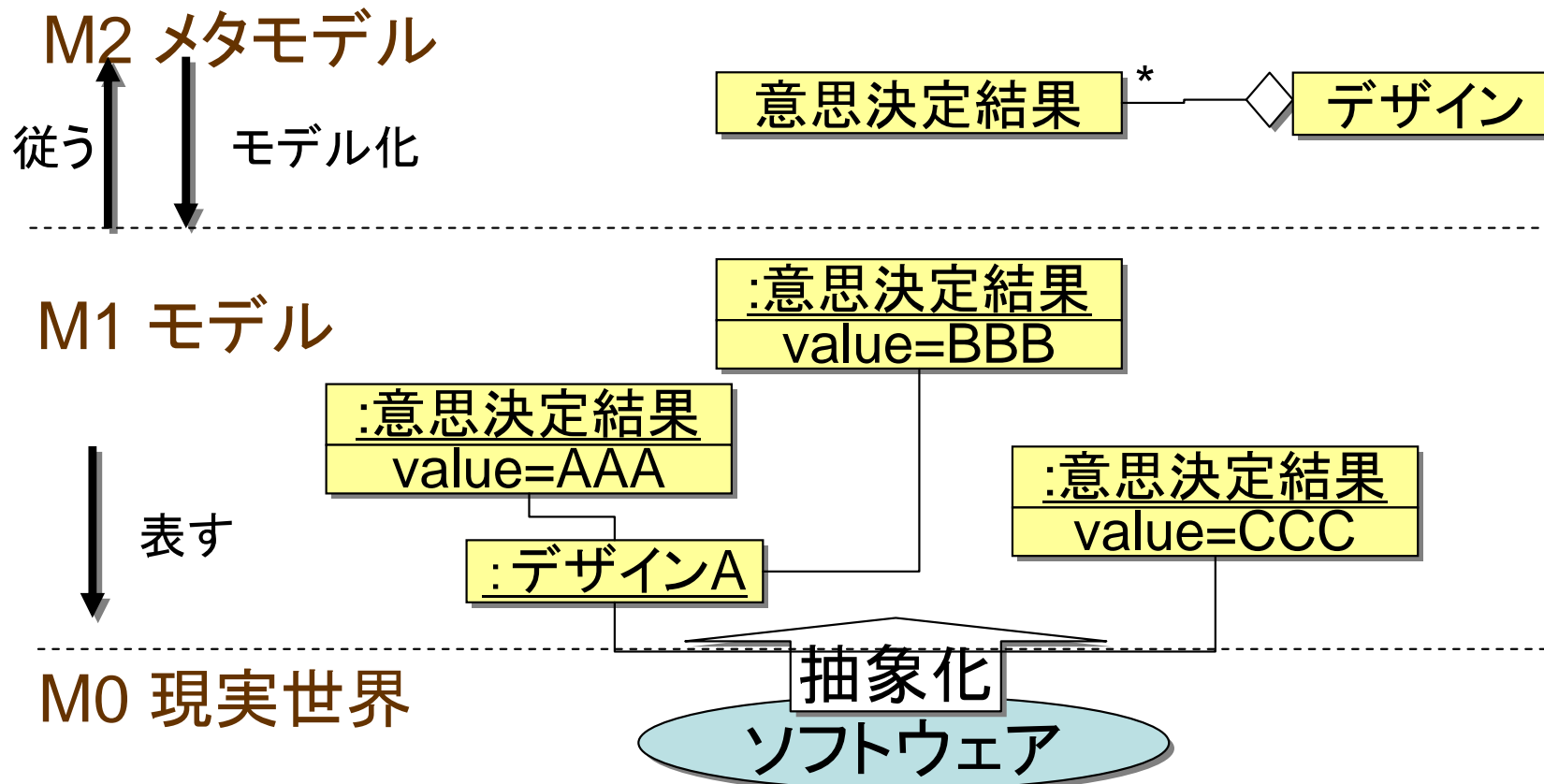
意思決定結果のモデル化(3/4)

要件(3)意思決定結果には, 細粒度なものから粗粒度まで様々な粒度がある.

➤例: Observerパターンを適用する。PullモデルでObserverパターンを適用する



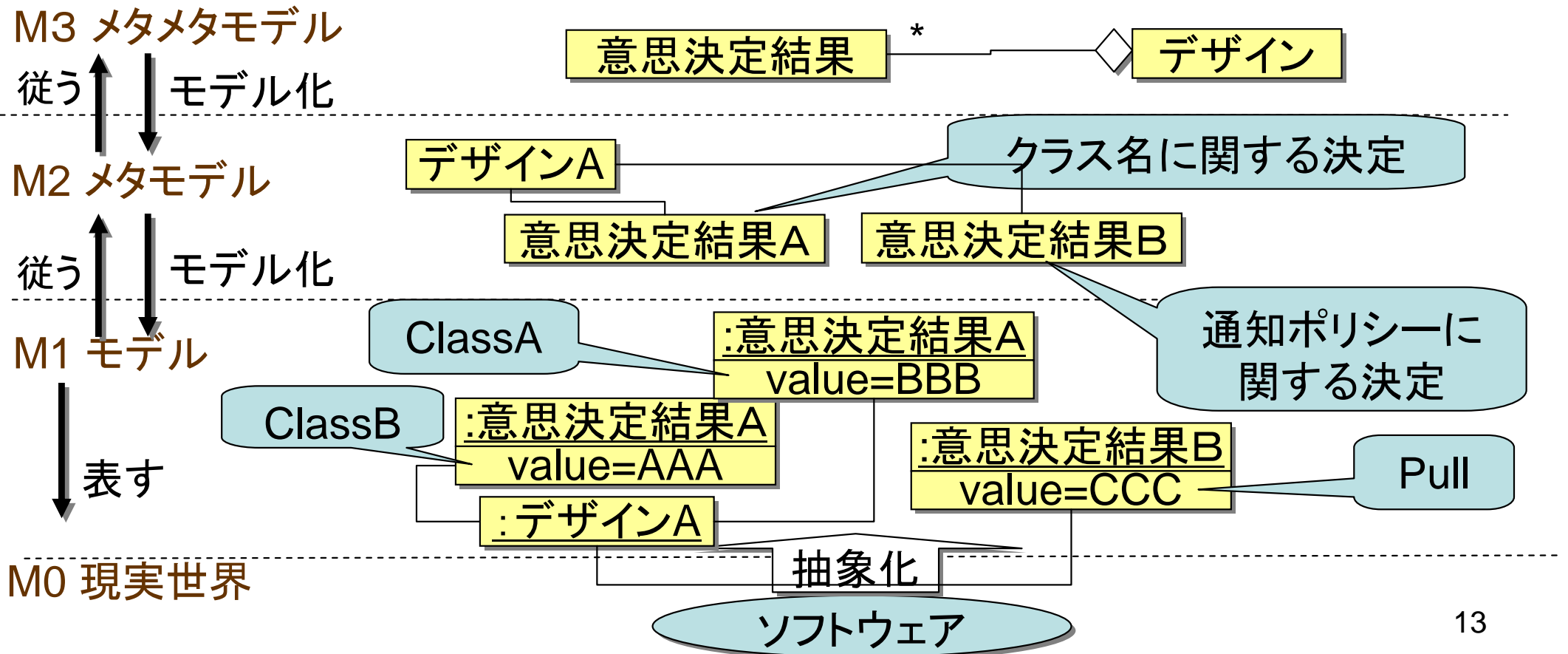
粒度を区別せずにモデル化



意思決定結果のモデル化(4/4)

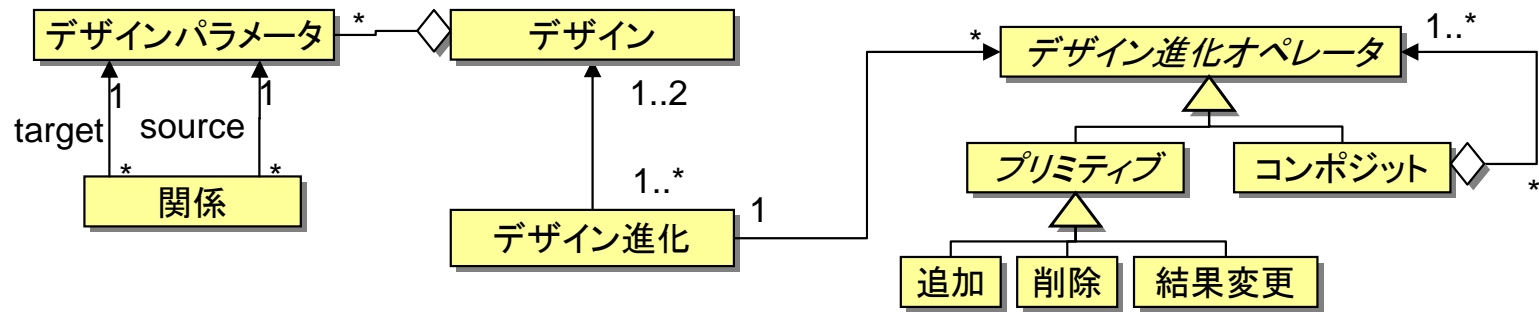
要件(4)意思決定結果には、同じ種類のものが存在する
➤例:クラス名に関する決定, 各Observerパターンインスタンスでの通知ポリシーに関する決定

- ・メタモデル層とモデル層の間にもう一層を追加して表現
- ・ある意思決定結果は、一つの意思決定結果の種類に属するとして表現 → オブジェクト指向におけるインスタンスとクラスの関係

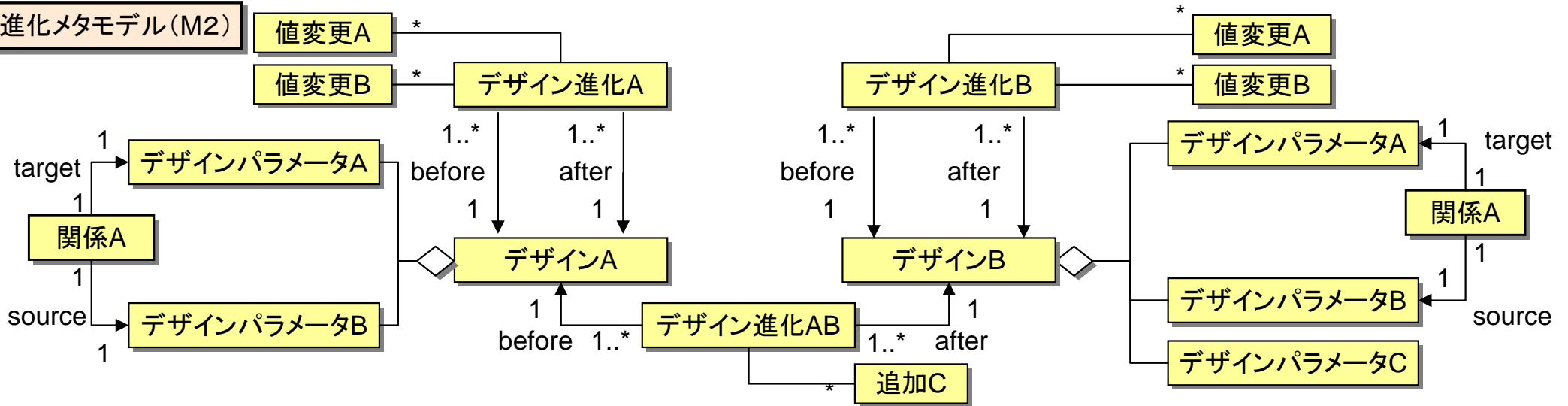


ソフトウェア進化のメタモデル階層

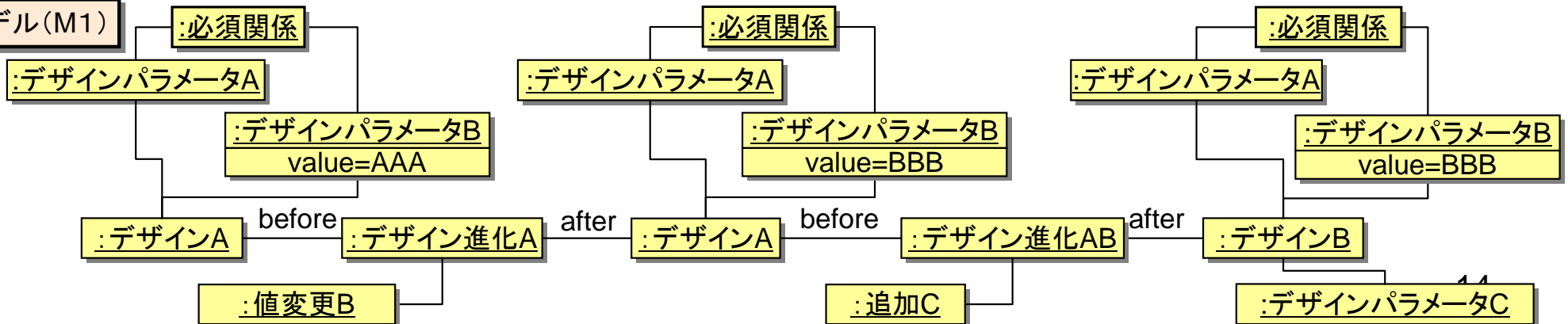
進化メタモデル (M3)



進化メタモデル (M2)



進化モデル (M1)



メタモデル階層の意味

各層の意味

- **進化メタメタモデル(M3)**: ソフトウェア進化をモデル化するためのモデリング言語を表す
 - 意思決定結果の種類, 意思決定結果の種類間の関係, 意思決定結果の種類に対する変更操作をモデル化
- **進化メタモデル(M2)**: 任意の抽象度や粒度でソフトウェア進化を表現するモデル定義を表す
- **進化モデル(M1)**: ソフトウェアの変化過程を表す

今後の課題

- ユーザ要求や環境要素のモデル化の検討
- 選択されなかった代替案としての意思決定結果のモデル化
- 例題によるモデルの妥当性と有効性の検証

まとめ

研究目標

- 開発者の行う意思決定を支援し、進化を方向付ける指針の開発

課題

- 意思決定結果とその変化の観点からのソフトウェア進化のモデル化

提案

- 3層から成るソフトウェア進化のメタモデル階層
 - 提案モデルを用いて進化の指針を開発

議論：メタモデル階層の意味

各層の意味

- M3: 進化を表現するためのモデリング言語を定義
- M2: 設計空間と, 空間探索のための操作, 設計空間の変化を定義
- M1:
 - 設計空間内の位置と空間内の移動の過程を表現

質問/議論

- デザインだけにかたよったモデル
 - デザイン進化がどこで始まってどこで終わるのが分からない
 - 要求の部分のモデル要素が必要
- 二層だと何が問題？
- フォーマルに進化を表現できて何がうれしいのか
- 変化過程を表現できるだけでは不十分
 - パラメータとオペレータ間を関係付けるような要素が必要？
 - どのオペレータを適用するか決定には役に立たない
- コンポーネント分割のプロセスを表現できた
 - M2モデルの定義
 - 関係の要素は、モデル化しにくい → いくつか共通の関係をモデル化しておくべきかも？
 - M1モデルの定義
 - 時間がなかったため、鹿糠さんの表記法で確認 → M1モデルも実際に定義して、その後に特定の表記法で表現するほうが安全？
- 前提条件/モデル化要件についての議論はなし

思ったこと

- 指針自体のモデルが必要？
 - パターンやリファクタリングを一般化したモデル？
- 設計原則と指針の違いや互いの関係
- 探索プロセスのモデルがあったほうがいい？
 - 畑村さんの決定学の研究を参考に？
- メタモデリングのツールがあると便利

- 設計空間のモデル化の研究

- モデル化の要件間の関係の整理
 - カテゴリわけ
 - 図で表現